

УПРАВЛЕНИЕ МОДЕЛЮ СИСТЕМЫ МАССОВОГО ОБСЛУЖИВАНИЯ С ИСПОЛЬЗОВАНИЕМ ДИНАМИЧЕСКИХ ПРИОРИТЕТОВ

Полянский С.В.

Научный руководитель - Ю.Я. Кацман, к.т.н., доцент

Томский политехнический университет, г.Томск

sergey2@tpu.ru

Введение

Система массового обслуживания (СМО) – динамическая система, целью которой является обслуживание случайного потока заявок при ограничениях на ресурсы системы.

В работе представлена разработанная и частично модифицированная имитационная модель системы массового обслуживания (СМО) [1]. Разработка модели осуществлялась в среде MatLab Simulink с использованием библиотек SimEvents и Stateflow.

Для модели СМО были установлены следующие ограничения:

- три источника поступления заявок;
- блок вывода необслуженных заявок при заполнении очереди;
- очередь;
- один обслуживающий прибор;
- время поступления заявок в систему распределено по равномерному закону;
- время обслуживания заявок распределено по экспоненциальному закону;
- безприоритетные $T_{ср} = 10$;
- средний приоритет $T_{ср} = 5$;
- абсолютный приоритет $T_{ср} = 3$;
- время моделирования – 1000 ед.;

Так как, согласно дисциплине обслуживания безприоритетные заявки поступают на обслуживающий прибор только при отсутствии в очереди заявок с относительным и абсолютным приоритетами, возможна такая ситуация, когда процент обслуженных заявок низкого приоритета становится пренебрежимо малым. Если исключение их из потока обслуженных заявок не допускается, возникает задача повышения процента обслуженных безприоритетных заявок. Эту задачу можно решить с помощью динамического увеличения приоритета, если поток безприоритетных заявок ниже порогового значения.

Подсистема очереди и динамического повышения приоритета

В работе [2] повышение приоритета безприоритетной заявки производилось, если время ее пребывания в очереди превышало допустимое. При использовании такого метода требовалось проведение предварительных тестовых расчетов, что не всегда удобно. Поэтому подсистема очереди и повышения приоритета была переработана.

Блок очереди состоит из блока Input Switch, Output Switch, систем коммутирования и повышения приоритета и трех блоков FIFO queue, по одному блоку для заявок каждого типа: FIFO queue для безприоритетных заявок, для заявок с относительным приоритетом и для заявок с абсолютным приоритетом.

После того, как заявка сгенерирована, она попадает в соответствующую очередь. Блок Input Switch в связке с системой коммутирования обеспечивает работу всего блока, как приоритетной очереди. Система коммутирования проверяет наличие заявок в очередях более высокого приоритета и если эта очередь не пуста, то коммутрует ее на выход из блока, т.е. сначала система проверяет наличие заявок в FIFO очереди для заявок абсолютного приоритета, если очередь не пуста, тогда блок Input Switch коммутрует эту очередь с выходом, если пуста, идет проверка FIFO очереди заявок с относительным приоритетом и так далее по аналогии. Если же коммутирована очередь с более низким приоритетом и в этот момент число заявок в очереди с более высоким приоритетом перестало равняться нулю, система коммутирования переключает Input Switch, коммутуруя очередь с более высоким приоритетом.

Система повышения приоритета работает в связке с блоком Output Switch.

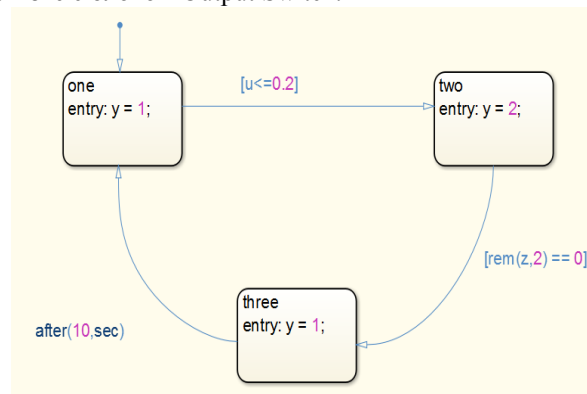


Рис. 1. Система повышения приоритета

Система повышения приоритета (см. Рис. 1.) проверяет процент обслуженных безприоритетных заявок, если процент ниже порогового значения, то данная система коммутурует очередь безприоритетных заявок с очередью заявок с относительным приоритетом, перекидывая две заявки из безприоритетной очереди в очередь заявок с относительным приоритетом. После этого, система не сразу повторно проверяет процент

обслуженных заявок, а через 10 единиц модельного времени, учитывается инерционность СМО. Если же процент обслуженных безприоритетных заявок достаточно высок и нет необходимости повышать вероятность обслуживания безприоритетной заявки, тогда данный блок коммутируется с блоком Input Switch.

Обсуждение результатов

В ходе экспериментов система находилась в перегруженном состоянии, такое состояние дает возможность оценить способность системы справляться с входящим потоком заявок. Для исследования характеристик СМО и проверки корректности работы системы динамического повышения приоритета был разработан набор тестов. Частичные результаты представлены на рис. 2.



Рис. 2. Количество обслуженных заявок

Система повышения приоритета включается, когда процент обслуженных безприоритетных заявок $< 20\%$. Из рисунка 2 видно, что в первых

трех опытах большую часть обслуженных заявок составляют заявки с относительным и абсолютным приоритетами, из-за этого возникает необходимость использования системы повышения приоритета. При использовании динамических приоритетов, процент обслуженных безприоритетных заявок достигал заданного уровня, но при этом снижалось количество обслуженных заявок относительного приоритета и общее количество обслуженных заявок, так как повышение приоритета не изменяет время обслуживания заявки, которое для безприоритетных заявок больше, чем у приоритетных. Количество обслуженных заявок с абсолютным приоритетом оставалось неизменным.

Заключение

Полученные результаты свидетельствуют о том, что в системе отсутствуют потери высокоприоритетных заявок за исключением случаев, когда высокоприоритетная заявка находится в обслуживающем приборе на момент окончания времени моделирования.

С помощью динамических приоритетов однократно повышается приоритет безприоритетной заявки, при этом вероятность обслуживания потока таких заявок увеличивается. При этом количество обслуженных заявок с относительным приоритетом уменьшается. Более того, такая заявка, имея приоритет, сохраняет среднее время обслуживания безприоритетной заявки. А так как безприоритетные заявки обслуживаются дольше, чем приоритетные, то и количество обслуженных системой заявок уменьшится, то есть такая характеристика СМО, как вероятность обслуживания потока входящих заявок уменьшится. В опытах 4 и 5 включение системы повышения приоритетов не изменило характеристики СМО, так как поток обслуженных безприоритетных заявок превышал пороговое значение, и динамические приоритеты не включались (рис. 2).

Список использованных источников

1. Katsman J.I., Apachidi X.N. Algorithm Simulation of Resource Allocation of the Queueing Systems, Based on the Priorities. Proceeding of 2014 International Conference on Mechanical Engineering, Automation and Control Systems, MEACS 2014, pp 1-6.
2. Apachidi X. N., Katsman Yu. Ya. Development of a queueing system with dynamic priorities. Key Engineering Materials. 685, 934-938